Bruno Scherrer, bruno.scherrer@inria.fr, March 2, 2023.

Consider a parity game G between EVEN (or Player 0) and ODD (or Player 1) on a state space $X = \{1, \ldots, n\}$ with priority function $\Omega : X \to \{1, \ldots, d\}$. On infinite plays, ODD wants that among the priorities that appears infinitely often, the maximal one is odd, while EVEN wants it to be even.

Let us make the game a bit more precise: a player that wins wants to do so with the highest possible priority, while a player that loses wants to do so with the lowest possible priority. This specification can for instance be dealt by Puri's reduction to a mean-payoff game. We shall call the function that assigns the resulting priority to each starting state the optimal parity function $p_* : X \to \{1, \ldots, d\}$.

Informally, we consider the following (recursive) algorithm that takes as input a game $\mathcal{G}$ of which the higher priority is p corresponding to Player $i = p \mod 2$, and return the optimal parity $p_* : X \to \{1, \ldots, d\}$.

1. In the game $\mathcal{G}$, compute the region A that can be won with priority p by Player i and the region B where Player $1-i$ can be sure that the hight priority occurring infinitely often is smaller than p.

2. Consider the game $\mathcal{G}'$ reduced to B. By definition of B, for any stationary policy of Player i, there exists a stationary policy for Player $1-i$ such that after at most n steps, the play only visit states with priority smaller that p. Compute the region C from which whatever Player i does, the play cannot reach a state with priority p (to play optimally, Player $1-i$ must force the play to reach this region).

3. Recursively solve the game $\mathcal{G}''$ reduced to C (it only involves states with parity smaller than p) and obtain its optimal parity function $q_* : C \to \{1, \ldots, p-1\}$.

4. On the game G', compute the optimal priority function r_* by propagating the parity function from C to B: among the policies for Player $1-i$ that prevent Player i from getting to any state of $B \backslash C$, compute the one that reaches a state $y \in C$ with the best priority $q_*(y)$ (against the best adversary played by Player i).

5. Return the optimal parity p_* where $p_*(x) = p$ if $x \in A$ and $p_*(x) = r_*(x)$ if $x \in B$.

Let us a give a more formal description. The sets A, B and C (steps 1 and 2) can be computed as follows. Consider the cost function

$$g(x) = \begin{cases} (-1)^i & \text{if } \Omega(x) = p, \\ 0 & \text{if } \Omega(x) < p. \end{cases}$$

Compute the solution of the n^2-horizon control problem with cost g and terminal cost 0, i.e. compute $v = T^{n^2} 0$ where the operator T is defined as follows for any w:

$$Tw(x) = \begin{cases} g(x) + max_{y \in \succ x} w(y) & \text{if } x \text{ is controlled by Player } i \\ g(x) + min_{y \in \succ x} w(y) & \text{if } x \text{ is controlled by Player } 1-i \end{cases}$$

Then

$$\begin{aligned} A &= \{x \; ; \; |v(x)| \geq n\} \\ B &= \{x \; ; \; |v(x)| < n\} \\ C &= \{x \; ; \; |v(x)| = 0\}. \end{aligned}$$

For the propagation (step 4), we consider the following terminal cost function

$$h(x) = \begin{cases} q_*(x) & \text{if } x \in C, \\ (-1)^i \infty & \text{if } x \in B \backslash C. \end{cases}$$

In this terminal cost function, the values on B are chosen to force Player $1-i$ to eventually terminate to any state of $B \backslash C$ (i.e. to be sure to terminate to some state of C). We finally compute $r_* = U^n h$ where the operator U is defined as follows for any w:

$$Uw(x) = \left\{ \begin{array}{ll} max_{y \in \succ x} w(y) & \text{if } x \text{ is controlled by Player } i \\ min_{y \in \succ x} w(y) & \text{if } x \text{ is controlled by Player } 1-i. \end{array} \right.$$